



Universidad Rey Juan Carlos



Escuela Técnica Superior de Ingeniería
Informática

Dirección y Gestión de Proyectos
Escuela de Informática del Ejercito

.. Estimación de proyectos software ..

Ejercicios



Grupo de Investigación Kybele



Departamento de Lenguajes y Sistemas
Informáticos II

Estos ejercicios son OPCIONALES, no requieren entrega.

Delphi

Ejercicio 1.

Tómese un programa sencillo ya realizado y su especificación. Es muy recomendable tomar un ejercicio de programación del que se tenga una solución totalmente codificada y su enunciado. Sobre dicho caso, realícese un Delphi normal para estimar el número de LDC.

Ejercicio 2.

Hágase lo mismo pero siguiendo la técnica Delphi de banda ancha.

Ejercicio 3.

Trabájese, de nuevo, con Delphi sobre el mismo caso pero ofreciendo como ayuda al menos otro enunciado parecido y el programa correspondiente o, en su defecto, el número de LDC generadas. ¿Qué pasaría si tuviéramos que estimar el plazo de tiempo para desarrollar el programa y nos dan como dato del tiempo consumido en el programa similar ya terminado?

Puntos Función

Ejercicio 4.

1. Obtener los Puntos de Función sin ajustar con la información disponible en la tabla:

Elemento	Simple	Promedio	Complejo	Total
Entradas	7	2	4	
Salidas	10	6	5	
Ficheros internos lógicos	12	0	0	
Ficheros de interfaz externos	0	0	12	
Consultas externas	5	6	4	
Puntos de Función sin ajustar				

2. Ajustar el valor obtenido a las características del proyecto. Para ello tener en cuenta estos requisitos no funcionales que nos ha brindado el usuario.

- La aplicación es en red
- Rendimiento y requisitos de diseño han sido definidos y revisados pero no requieren ninguna acción especial.
- La frecuencia de transacciones es semanal, es decir, el período pico de operación es semanal.
- Todas las entradas de datos son on-line
- Algunas funciones que ayudan a la eficiencia del usuario final
- Todas las actualizaciones de archivos son on-line
- Se utiliza código reusable dentro de la aplicación.
- No se realizaron consideraciones ni se requirieron desarrollos especiales para la instalación por parte del usuario.
- Se diseñó la aplicación para ser utilizada en múltiples lugares pero funcionará bajo entornos idénticos de hardware y software.
- La aplicación no dispone de cálculos complejos.

3. Estimar las Líneas de Código para C++, Java y Vb.

COCOMO

Ejercicio 5.

Se desea desarrollar el software para un controlador de presión del aire en cabina para el avión AIR-96. Residente en ROM dicho software interactuará con los mecanismos de ajuste en función de las señales que le llegan de los sensores. El tamaño estimado del sistema es de 10 KLDC. Por otro lado, las constantes mejoras a nivel hardware provocarán una gran volatilidad de los requisitos.

Respecto a los factores de coste, conviene tener en cuenta que se trata de un sistema muy complejo, que requiere una fiabilidad muy alta, la base de datos tiene un tamaño reducido y existen restricciones muy altas de memoria, tiempo de ejecución y tiempo de respuesta. Respecto a los programadores, podemos decir que tienen mucha experiencia en trabajar con este tipo de hardware (maquina virtual) y son de alto nivel.

- 1) Estímese el esfuerzo requerido en personas-mes y el tiempo de desarrollo necesario.

Ejercicio 6.

Es usted el jefe de proyecto en un proyecto para el desarrollo de un sistema de comunicaciones cuyo tamaño, partir de una especificación bastante detallada pero en la que aún no se dispone del diseño de bajo nivel, se ha estimado en 30 KLDC.

El software a desarrollar es de alta complejidad pero afortunadamente podremos contar con personal de muy alta cualificación y experiencia específica en el tipo de software a desarrollar. Teniendo en cuenta que el costo del salario mensual de cada persona es de 1350€ mensuales, se pide:

- 1) Calcule el esfuerzo necesario para el desarrollo de este sistema.
- 2) ¿Sería más rentable en tiempo y dinero emplear personas de nivel medio cuyo salario es de 1.275 € mensuales?

Ejercicio 7.

Se está planificando un nuevo proyecto que se desarrollará utilizando el lenguaje Java.

A partir de una especificación de requisitos bastante madura (pero aún por refinar) se ha estimado su tamaño en 261.36 PF.

Las características más destacadas del proyecto son:

- La naturaleza del sistema a desarrollar impone un nivel alto de fiabilidad, pues cualquier fallo implicaría pérdidas económicas importantes. Igualmente, el sistema debe ser capaz de responder a un alto número de solicitudes cuando se está ejecutando y debe hacerlo con rapidez pues tiene una alta carga de interacción con el usuario. No obstante, a pesar de las observaciones anteriores, se trata de un software poco complejo.
- Para su desarrollo se cuenta como programadores muy experimentados en este tipo de proyectos y con un nivel muy alto de capacitación en el lenguaje C#. La capacitación de los analistas es algo menor, pero también alta. Además, se cuenta con licencias para utilizar entornos de desarrollo altamente apropiados.
- Finalmente, al tratarse de un proyecto para uso interno, hay algo menos de presión de la habitual con respecto a las fechas de fin.

Se pide:

- 1) Utilizar el método de estimación de COCOMO para estimar el esfuerzo necesario para el desarrollo del proyecto
- 2) Obtener también la estimación del Tiempo de Desarrollo
- 3) Obtener la Productividad necesaria de cada participante para cumplir con las estimaciones realizadas

- 4) Estimar cuanto personal debería dedicarse al proyecto para cumplir con estas estimaciones
- 5) Suponiendo que el Tiempo de Desarrollo no debiese superar los 3 meses, ¿Cuánto personal asignaría para que el resto de estimaciones se mantuviesen igual?

Ejercicio 8.

Se planea abordar el desarrollo de una Aplicación para la gestión de clientes en lenguaje C estándar para una empresa de distribución que apoya su gestión en dos bases de datos:

- Datos de clientes de gran complejidad.
 - Un fichero de *back-up* de poca complejidad.
-
- El sistema soporta tres tipos de operaciones distintas: altas, bajas y modificaciones. Todas ellas de complejidad alta.
 - Así mismo, debe generar hasta tres tipos distintos de informes sobre clientes, los tres de complejidad media.
 - Además, para facilitar su integración con otros sistemas, la aplicación transfiere a otras aplicaciones dos ficheros de complejidad media para otros tantos sistemas (contabilidad y dirección por objetivos).
 - Por último, el sistema soporta dos consultas de complejidad baja y una consulta de ayuda, a plena pantalla, de gran complejidad.
 - El desarrollo del proyecto se realizará en un entorno cuyos factores de coste (complejidad) serán todos de tipo medio, excepto la entrada de datos *on-line* (valor 5), la actualización *on-line* (valor 5) y la facilidad de operación (valor 5).

- 1) Calcúlese las líneas de código necesarias para implementar la aplicación
- 2) Una vez calculadas las LDC necesarias en C, hállese también las LDC necesarias para implementar la aplicación en COBOL 85. Sobre este número de LDC en COBOL, calcúlese, siguiendo el modelo COCOMO, el esfuerzo en personas-mes en un entorno de trabajo de alto nivel de utilización de herramientas, de técnicas modernas de programación y con programadores de muy alta cualificación (analícese el resto de factores de coste; por defecto puede tomarse valor nominal).
- 3) También debe obtenerse el coste monetario del esfuerzo asumiendo un salario medio bruto anual (14 pagas) de 28.000€. por persona.
- 4) Por último, el jefe de departamento de informática debe decidir a quién de los tres miembros del equipo de desarrollo (Anacleto, Pilón y Fifi) concede un aumento de sueldo, basándose exclusivamente en su productividad. El proyecto finalmente se desarrolla en COBOL 85, en C y en un lenguaje de manejo de base de datos. Para ello, contabiliza las LDC generadas en la parte que ha realizado cada uno de ellos (que corresponde con lo codificado en cada uno de los tres lenguajes) y averigua lo siguiente:
 - En COBOL, Pilón ha generado 3.975 LDC para implementar las entradas y las salidas de la aplicación.
 - En lenguaje C, Fifi ha generado 2.007 LDC para implementar la generación de ficheros de intercambio con otras aplicaciones.
 - En lenguaje de base de datos, Anacleto ha generado 1536 LDC para implementar las bases de datos y las consultas.

Por lo tanto, el jefe decide que Pilón ha sido el más productivo. ¿Es la decisión más acertada? Razónese la respuesta.

Puntos Caso de Uso

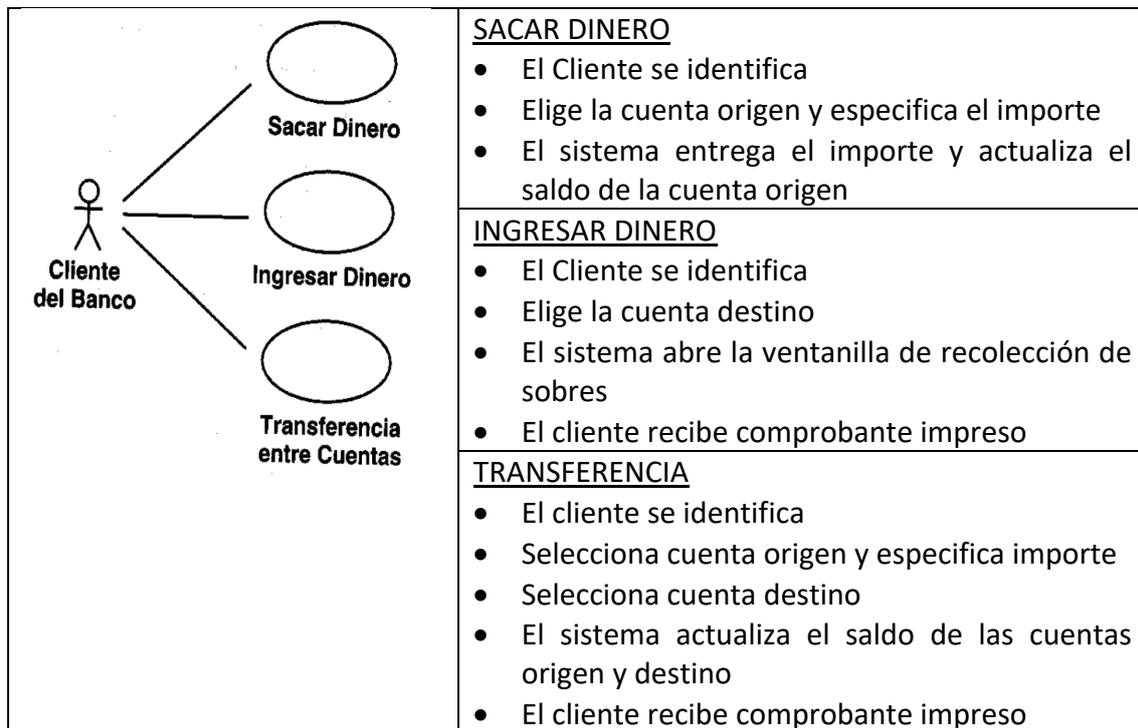
Ejercicio 9.

Utilizar la técnica de Puntos Caso de Uso para estimar el esfuerzo (horas) y la duración de un proyecto en el que se desarrolla el software de un cajero automático que incluye:

- Casos de Uso: Retirar dinero, Depositar dinero y Transferir dinero entre cuentas
- Actores: Cliente, Sistema bancario

Se pide:

- Calcular el tamaño de la aplicación en términos de Puntos Caso de Uso
- Considerar que es un proyecto nuevo para derivar la duración del proyecto suponiendo
 - Que es un proyecto nuevo (factor de productividad estándar: 20 h/UCP).
 - Que se dispone de un programador que trabaja 40 horas/semana.



Ejercicio 10.

Hasta ahora, la compañía TNC gestionaba los datos de sus clientes y las tarjetas de crédito de aquellos de forma manual. Para ello, el operador que se encarga de la recogida manual de datos, se encarga también de verificar los datos de la tarjeta de crédito, utilizando para ello un portal externo para pagos electrónicos. De cada cliente, la compañía guarda su código, nombre, dirección, teléfono y los datos de la tarjeta verificados externamente. TNC quiere automatizar la gestión de datos de acuerdo a las siguientes premisas:

- El sistema debe comprobar automáticamente que el código de cliente asignado es único.
- El código no debe tener más de 8 caracteres.
- La verificación de la información de la tarjeta debe ser automática. Para ello TNC ha proporcionado la API que permite interactuar con el sistema de verificación externo.
- El número de la tarjeta de crédito no debe tener más de 10 caracteres.
- El operador que se encarga de la entrada de datos debe poder añadir, modificar y eliminar información de clientes.
- La BBDD de clientes estará ubicada en la central de TNC y sólo los operadores podrán utilizar la aplicación para gestión de datos que se debe desarrollar.

- Dicha aplicación debe correr sobre Windows. En particular, todas las máquinas de TNC tienen instalado Windows 7.

Esta especificación se recoge en el siguiente Caso de Uso:

Código del UC	DATAENTRYPROJECT-19
Nombre del UC	Mantener datos de clientes
Descripción	Este caso de uso recoge la funcionalidad de la aplicación para el mantenimiento de clientes
Actores	Operador de entrada de datos
Disparador	El operador pulsa el botón "Añadir datos de cliente"
Precondiciones	<ul style="list-style-type: none"> • El operador tiene acceso a Internet • El operador se ha logueado en el sistema
Supuestos	Los datos de cliente se introducen manualmente. No se considera mecanismo de importación automático.
Condiciones de error	<ul style="list-style-type: none"> • Los datos del cliente no son añadidos correctamente a la BBDD y se muestra el correspondiente mensaje de error • El código de cliente ya existe en la BBDD • El código sobrepasa la extensión máxima • El número de tarjeta sobrepasa la extensión máxima • La verificación de la tarjeta de crédito por el sistema externo falla
Flujo Principal	<p>AÑADIR CLIENTE</p> <ol style="list-style-type: none"> 1. El operador recibe los datos del cliente 2. El operador introduce los siguientes datos: <ol style="list-style-type: none"> a. Código de cliente b. Nombre c. Dirección d. Teléfono 3. El sistema comprueba si el código de cliente ya existe en la BBDD <ol style="list-style-type: none"> a. Si existe se eleva el error "Código de cliente duplicado" b. Si el código sobrepasa la extensión máxima se eleva el error "Código demasiado largo" 4. El operador introduce los datos de la tarjeta. <ol style="list-style-type: none"> a. Si el número sobrepasa la extensión máxima se eleva el error "Número de tarjeta demasiado largo" 5. La información de la tarjeta se envía al sistema externo de verificación (utilizando la API proporcionada). 6. El sistema de verificación envía confirmación del resultado proceso de verificación ("OK / Datos NO Válidos") 7. El operador añade los datos del cliente a la BBDD
Flujo Alternativo (extensión)	<p>MODIFICAR CLIENTE</p> <ol style="list-style-type: none"> 1. El operador introduce el código del cliente cuyos datos deben ser actualizados. 2. El sistema recupera y muestra los datos del cliente. 3. El operador realiza las modificaciones apropiadas. 4. Se repiten los pasos 1 a 6 del Flujo AÑADIR CLIENTE 5. El operador actualiza los datos del cliente en la BBDD

Flujo Alternativo (extensión)	<u>ELIMINAR CLIENTE</u> <ol style="list-style-type: none">1. El operador introduce el código del cliente cuyos datos deben ser eliminados.2. El sistema recupera y muestra los datos del cliente.3. El sistema muestra el mensaje “¿Está seguro de que quiere eliminar los datos del cliente?”<ol style="list-style-type: none">a. Si el operador pulsa “Si”, los datos del cliente se borran de la BBDD.b. Si el operador pulsa “No”, el sistema no lleva a cabo ninguna acción.
Post-condiciones	<ul style="list-style-type: none">• Los datos del cliente son añadidos a la BBDD• Los datos del cliente son actualizados en la BBDD• Los datos del cliente son borrados de la BBDD
Consideraciones adicionales	Si la API del sistema de verificación externo se viese modificada, la interacción del sistema con el sistema de verificación deberá ser actualizada consecuentemente.

Se pide:

- Calcular los UUCP (Puntos Caso de Uso no Ajustados)
- Calcular los UCP

De cara a calcular los UCP ténganse en cuenta los siguientes factores técnicos

- El sistema, que seguirá una arquitectura simple de dos capas, no realiza en realidad ningún procesamiento complejo más allá de la verificación de las extensiones de los campos pues la verificación de los datos de la tarjeta recae sobre el sistema externo.
- En general, los operadores que utilizarán la aplicación están muy habituados a utilizar este tipo de aplicaciones. Por tanto, la velocidad de respuesta es importante ya que los operadores deben ser capaces de cargar datos sin introducir tiempos de espera intermedios.
- En principio se trata de un proyecto pequeño y aislado para el que no se prevén modificaciones posteriores.
- La aplicación se desarrollará en C#, con lo que se aprovecharán las facilidades para la creación de instaladores que ofrece la plataforma .NET.

... y las siguientes características del proyecto:

- Es un proyecto muy simple y sencillo con lo que no hay necesidades especiales en cuanto a metodologías, experiencia del analista o en este tipo de desarrollos.
- La mayoría de los desarrolladores de la empresa (que trabajan a tiempo completo) están familiarizados con la programación orientada a objetos, pero no han desarrollado nunca con C#.
- Considerar que es un proyecto nuevo (factor de productividad: 20-28 h/UCP) y derivar el coste del proyecto sabiendo que se dispondrá de un programador cuyo salario es 1600 €/mes.